

*Матеріали VII Міжнародної науково-технічної конференції молодих учених та студентів.**Актуальні задачі сучасних технологій – Тернопіль 28-29 листопада 2018.***УДК 004.43****А.М. Луцків канд.техн.наук, доц., А.В. Цапко**

Тернопільський національний технічний університет імені Івана Пулюя, Україна

КОНЦЕПЦІЇ СХОВИЩ ДАНИХ У КОНТЕКСТІ МІГРАЦІЇ З SQL НА NOSQL**A.M. Lutskev (Ph.D.; Assoc. Prof.), A.V. Tsapko****CONCEPTS OF DATA WAREHOUSES IN THE CONTEXT OF MIGRATING FROM SQL TO NOSQL**

Бази даних становлять невід'ємну складову інформаційної інфраструктури сучасних підприємств та організацій. Останні роки спостерігається чітка тенденція до зростання даних у цих сховищах та, відповідно, кількості задач з їх опрацювання. Потреба в розв'язанні цих задач породила появу цілої низки нових платформ, інструментів для великих обсягів різноманітних та неструктурованих даних. У даному випадку на противагу відомим реляційним сховищам даних з'являються нереляційні, або NoSQL (з англ. Not only SQL). Як впливає з назви, NoSQL пропонує певну концепцію на противагу домінуючій довгий час парадигмі SQL. Тому перехід з реляційного сховища на нереляційне передбачає не просто міграцію даних, а й перегляд концепції їх опрацювання та їх моделі.

SQL є структурованою мовою запитів, яка базується на реляційній моделі бази даних (RDMS) [1]. Будь-яке представлення даних зводиться до сукупності двовимірних відношень, які часто, для спрощення розуміння, подаються у вигляді таблиць. Ці відношення, а також інші елементи реляційних баз даних (кортежі, домени та ін.) базуються на математичному апараті реляційної алгебри (англ. relation). Реляційною вважається БД, у якій всі дані, які доступні користувачеві організовані у вигляді набору двовимірних відношень (таблиць), а всі операції над даними зводяться до операцій над цими таблицями. Реляційні оператори мають одну важливу властивість: вони замкнені відносно поняття відношення. Це означає, що вирази реляційної алгебри визначаються над відношеннями реляційних БД і результатом обчислення також є відношення. Набір основних алгебраїчних операцій складається з восьми операцій, які діляться на два класи - теоретико-множинні операції та спеціальні реляційні операції, доповнені деякими спеціальними операціями, специфічними для баз даних [2]. До складу теоретико-множинних операцій входять традиційні операції над множинами: об'єднання, перетин, різниця, декартовий добуток. Спеціальні реляційні операції включають: вибірку, проєкцію, природне об'єднання, поділ.

NoSQL для роботи з колекціями не використовує згаданих підходів, або ж використовує їх частково (залежно від реалізації бази/сховища даних). Операції між колекціями не обов'язково створюють іншу колекцію. Відсутні, також, загальні операції для NoSQL [3]. Загалом, сімейство нереляційних баз даних є доволі різноманітним й визначається сферою застосування й базується на природі тих даних, які мають зберігатись у сховищі й опрацьовуватись ним. А тому, концепції та моделі які використовуються в основі роботи нереляційних баз даних, можуть бути доволі різноманітними: графові бази даних (мат. апарат графів), сховища ключ-значення (асоціативні масиви великих розмірів), стовпцево- та рядково-орієнтовані й інші.

Обидва типи БД мають свої концепції транзакційності: для SQL – це ACID, NoSQL — CAP. ACID (Atomicity, Consistency, Isolation, Durability) є аббревіатурою чотирьох основних атрибутів:

- Atomicity (атомарність). Жодна транзакція не буде виконана частково. Будуть або виконані всі операції, що беруть участь у транзакції, або не виконано жодної.

- Consistency (Узгодженість). Транзакція створює новий, дійсний стан даних або, якщо виникає будь-який збій, повертає всі дані до стану перед початком транзакції.

- Isolation (Ізольованість). Транзакція, яка ще не виконана, повинна залишатися ізольованою від будь-якої іншої транзакції.

- Durability (Довговічність). Система, що зберігає дані, зберігає їх таким чином, що навіть у разі виходу з ладу і перезавантаження системи дані будуть доступні у правильному стані.

Концепція ACID описана у розділі ISO/IEC 10026-1: 1992, розділ 4[4]. У контексті ізольованості варто також зважати на рівні ізоляції, які задаються у реляційній базі даних й визначають роботу з даними: Serializable (впорядкованість), Repeatable read (повторюваність читання), Read committed (читання фіксованих даних), Read uncommitted (читання незафіксованих даних).

Оскільки, сховища NoSQL є розподіленими, то забезпечення ACID не є можливим. У даному контексті використовується теорема CAP[5], яка стверджує, що розподілені мережеві сховища та системи опрацювання даних можуть лише гарантувати або підтримувати дві з трьох наступних властивостей:

- Consistency (узгодженість) - усі вузли бачать однакові дані у будь-який момент часу;

- Availability (доступність) - гарантія того, що кожен запит отримає коректну відповідь;

- Partition Tolerant (стійкість до розподіленості) - не зважаючи на розподіленість даних або, можливі, втрати зв'язку з частиною вузлів сховища, система стабільно працює і здатна коректно відповідати на запити.

Тобто намагаються дотримуватись певного балансу між доступністю, узгодженістю та розподіленістю: наявні гарантовані властивості CA, AP або CP. CAP теорема базується на принципі узгодженості в кінцевому випадку. Частинним випадком CAP теореми є теорема PACELC: у випадку мережевої розподіленості (P - network partitioning) у розподіленій комп'ютерній системі необхідно обирати між доступністю (A — availability) та консистентністю (C — consistency) (згідно з теоремою CAP), але ще (E - else), навіть, коли система працює нормально за відсутності розподілу даних, потрібно обирати між латентністю (L - latency) і узгодженістю (C - consistency).

При міграції з реляційного на нереляційне сховище даних перед інженером постає задача врахування згаданих парадигм, а також особливостей використовуваних мережевих комп'ютерних систем: латентності та пропускну здатності мереж передавання даних, потужностей обчислювальних вузлів сховищ даних, логічної розподіленості даних і взаємозв'язків між ними.

Література

1. Eck R. NoSQL vs SQL: what you need to know/Ralph Eck// [Електронний ресурс] Режим доступу: URL: <http://www.monitis.com/blog/nosql-vs-sql-what-you-need-to-know/>
2. Date C. SQL and Relational Theory, 3rd Edition / Chris Date // O'Reilly Media, 2015, 582p.
3. Varga V. Conceptual Design of Document NoSQL Database with Formal Concept Analysis/ Viorica Varga, Katalin Tünde János-Rancz, Balázs Kálmán // Acta Polytechnica Hungarica, Vol. 13, No. 2, 2016 [Електронний ресурс] Режим доступу: URL: https://www.uni-obuda.hu/journal/Varga_Janos-Rancz_Kalman_66.pdf
4. Celko J. Joe Celko's SQL for Smarties, 3rd Edition / Joe Celko// Morgan Kaufmann, 2010 [Електронний ресурс] Режим доступу: URL: <https://www.oreilly.com/library/view/joe-celkos-sql/9780123693792/>
5. Mehra A. Understanding the CAP theorem /Akhil Mehra//DZone. Database Zone , 2017 [Електронний ресурс] Режим доступу: URL: <https://dzone.com/articles/understanding-the-cap-theorem>